Modeling Symmetric Developable Surfaces from a Single Photo

Amélie Fondevilla, Adrien Bousseau, Damien Rohmer, Stefanie Hahmann, Marie-Paule Cani

Journées Françaises d'Informatique Graphique,
Grenoble, 30 nov - 2 dec 2016

Modeling Symmetric Developable Surfaces from a Single Photo

Amélie Fondevilla[1], Adrien Bousseau[2], Damien Rohmer[1,3], Stefanie Hahmann[1], Marie-Paule Cani[1]

[1] Univ. Grenoble Alpes & CNRS (LJK), Inria , [2] Inria Sophia-Antipolis , [3] Univ. Lyon, CPE Lyon

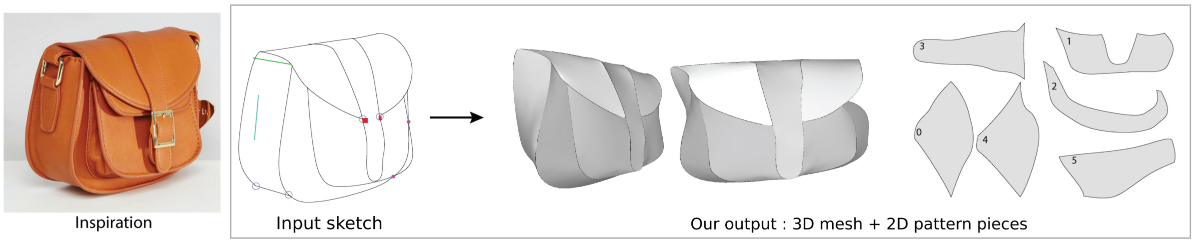

Figure 1: Our method reconstructs 3D mesh and 2D pattern pieces of a sewed object from a single annotated sketch.

Abstract

We propose a method to reconstruct 3D developable surfaces from a single 2D drawing traced and annotated over a side-view photo of a partially symmetrical object. Our reconstruction algorithm combines symmetry and orthogonality shapes cues within a unified optimization framework that solves for the 3D position of the Bezier control points of the drawn curves while being tolerant to drawing inaccuracy and perspective distortions. We then rely on existing surface optimization methods to produce a developable surface that interpolates our 3D curves. Our method is particularly well suited for the modeling and fabrication of fashion items as it converts the input drawing into flattened developable patterns ready for sewing.

1. Introduction

We describe a method to reconstruct the 3D shape of a developable sewed object from a drawing traced and annotated over a side-view photo. The main challenges are to overcome the inherent ambiguity caused by a single-view input and to model shapes that are piece-wise developable so that it is possible to output flattened pattern pieces ready for fabrication of the object. The main contribution of our approach is a new formulation for 3D curve reconstruction that combines partial symmetrical information with orthogonality cues to lift the curves in 3D. This formulation also includes a regularization term that balances shape distortions over reprojection error to account for drawing inaccuracy, perspective effects, and partially visible symmetries. We integrate our 3D curve reconstruction into a complete pipeline that generates 3D piecewise developable surfaces and output the corresponding flattened patterns for real-world sewing. Let us consider a single photo representing a partially symmetrical shape from a side view assumed to be seen closed from an orthographic projection, as shown in Figure 1. Our system takes as inputs the 2D information of silhouette, seam and border curves provided as user annotations on top of such photo, as well as extra information such as orthogonality and symmetrical key points as explained in Sec. 3.1. Our algorithm works in three steps. First, partially symmetrical curves are automatically extracted from the input data and a point-to-point correspondence is computed as explained in Sec. 3.2. Second, the depth information of the curves is computed by solving an optimization system considering the annotated data as soft constraints as explained in Sec. 3.3. Third, 3D piecewise quasi-developable surfaces and the associated flattened patterns are synthesized from the set of 3D curves as explained in Sec. 3.4.

2. Related work

Optimisation-based reconstruction. Various methods for sketch-based modeling from single-view have been proposed, as reported in [OSSJ09]. Many of them are formulating the recovery of third dimension with mathematical constraints, performing the reconstruction by solving an optimization problem on these constraints.
Some methods are dealing with drawings made of straight lines, representing polygonal shapes with planar faces [LS07], [LFG08]. These approaches rely on constraints that are specific to architectural structures, such as face planarity and cubical corners. Single-view reconstruction of smooth surfaces is a more complex issue as the above constraints do not apply. Specific curves called cross-sections have been recently used to recover normal map [SBSS12], and then 3D surface [XCS*14] from a single hand-drawn sketch of curves network. The cross-sections are planar curves intersecting orthogonally on the surfaces, carrying therefore underlying 3D information. They are commonly drawn in en-

gineering design sketches, but unfortunately not in fashion. Our method inspires from these approaches in using orthogonal and smoothness curve constraints, but does not rely on cross-sectional curves, allowing for more general input. Instead, we introduce other constraints such as symmetry to better fit fashion item designs.

Inferring 3D from symmetry. Most of fashion garments and accessories are partially symmetric, which represents a strong constraint for the reconstruction. The approach presented in [ÖUP*11] recovers the depth of a hand-drawn symmetric sketch requiring the user to annotate a few points in the sketch lying on the symmetry plane. The algorithm infers the orientation of the symmetry and then performs the matching between the curves in the sketch, based on feature points computed in each curves. On the other hand, Cordier *et al.* [CSMS13] presented a method to compute the symmetry orientation without any user annotations. The orientation of the symmetry is chosen as the one that maximizes the amount of possible pairs of symmetrical curves, and the compactness of the resulting 3D shape. In both methods, the reconstruction is fully determined by the symmetry assumption, without requiring other optimization. Even if these methods are applicable to sketches containing approximate symmetry, they however require the curves to represent the full shape of the object, while our sketches are generally not representing the occluded parts of them. Moreover, these approaches require all the curves of the drawing either to have a corresponding symmetrical curve, to be self-symmetric or to lie in the symmetry plane. The remaining curves are generally left as planar. Our method therefore inspires from these approaches, but will use the symmetry as a soft constraint in a global optimization process.

3. Developable surface reconstruction algorithm

Our approach combines the benefits of optimization-based reconstruction and the strength of the assumption of mirror-symmetry to robustly handle partially symmetrical shape reconstruction. We present a whole modeling pipeline which provides a piecewise developable surface from a single annotated photo. We will first use the annotated photo to reconstruct a network of 3D curves. These curves are then used to synthesize a piecewise developable mesh representing the surface of the sewed object.

3.1. Input data and user annotations

The inputs of our program are 2D points and curves provided as an SVG file as illustrated in Fig. 1. The curves are defined as cubic Bezier polynomials representing the seams, borders, and silhouettes of the object, along with other point and segment annotations pointing out distinctive geometric features such as symmetrical key points, orthogonal intersection, and symmetry plane axes. Note that in our case, these annotations were drawn on top of a real photo assumed to be close to orthographic projection using the vector graphics software Inkscape. More precisely, we distinguish 4 different annotations :

- pairs of partially symmetrical curves, drawn in black in our sketch.
- two symmetrical vertices in the image, providing the 2D direction of the line of symmetry (linking all pairs of symmetric points), shown as red squares in our sketch.
- two vectors representing the 2D projection of two arbitrary orthogonal vectors lying in the symmetry plane : this information, along with the line of symmetry allows us to compute the 3D normal vector of the symmetry plane and is shown as the green and cyan segments in our sketch.
- all intersections that are orthogonal in 3D as shown by the blue circle in our sketch.

We assume for the reconstruction that the input sketch is representing a *non-accidental* view of the object, in the sense that slight changes in viewpoint do not lead to large changes in the 3D interpretation.

3.2. Symmetrical point matcher algorithm

As stated previously, the partially symmetrical curves are described in our 2D input as Bezier curves. We also get as an input the direction of the line of symmetry for all the pairs of symmetrical curves. Note that these inputs do not provide directly full point-wise correspondence for the symmetrical parts of the curves. Our first step consists therefore to compute such correspondence to extract which part of the 2D curves are symmetrical each other and handle the partial visibility of some inputs. To this end, we propose an algorithm to match pairs of symmetric points within two sampled 2D curves. This algorithm will be used in the reconstruction of 3D curves (Sec 3.3). A sampled curve C_m is represented as an ordinate set of 2D points $\{V_k^m\}_k$.
Our algorithm takes as an input two sampled curves (C_m, C_n) along with the line of symmetry $s \in \mathbb{R}^2$, which corresponds to the normalized orthographic projection of the normal vector of the 3D symmetry plane. The line of symmetry is collinear to each line linking a pair of symmetrical points in 2D. The algorithm outputs the set of symmetrical correspondences between C_m and C_n :

$$S_{m,n} = \left\{(V_k^m, V_l^n) \in C_m \times C_n | V_k^m, V_l^n \text{are symmetrical}\right\} \quad (1)$$

Symmetrical likelihood. Two points are considered as symmetrical if the 2D vector linking them is sufficiently close to the symmetry line $\vec{s}$. We define the symmetrical likelihood for two sampled points V_k^m, V_l^n as

$$\mathcal{L}(V_k^m, V_l^n, \vec{s}) = 1 - \tfrac{1}{\pi}\min\left(\alpha_{\vec{s}}(V_k^m, V_l^n), \alpha_{\vec{s}}(V_l^n, V_k^m)\right),$$

where $\alpha_{\vec{s}}(V, V') = \left|\arccos\left(\frac{\overrightarrow{VV'}}{||\overrightarrow{VV'}||}.\vec{s}\right)\right|$.
We compute a first set of symmetrical correspondences :

$$\begin{aligned} S'_{m,n} = \{(V_k^m, V_l^n)\,|\quad & V_l^n = \underset{V}{\operatorname{argmax}}\left(\mathcal{L}(V_k^m, V, \vec{s})\right) \\ & \text{and } \mathcal{L}(V_k^m, V_l^n, \vec{s}) > K\} \end{aligned} \quad (2)$$

A one-to-one matching between the points is not always possible, nor wanted, because the curves may not contain the same amount of vertices, and the symmetry may only be partial. We thus introduce a user defined threshold of acceptance K for two points in the curves to be considered as symmetrical. Note that in general $S'_{m,n} \neq S'_{n,m}$ since the amount of symmetrical correlation is limited by the amount of sampled points in each curve. Thus, we compute $S'_{m,n}$ where $|C_m| \leq |C_n|$, and then, remove from $S'_{m,n}$ the potential multiple occurrences of any point V_l^n.

Consistency. The 2D representation of a 3D symmetry may contain ambiguous cases, as in the example of Figure 2, and may not be trivial to solve as explained in [CSMS13]. We propose a method which computes a consistent set of symmetrical correlation for two symmetrical curves. To discard unwanted correspondences, we partition $S'_{m,n}$ into subsets containing pairs of points that have a continuous middle

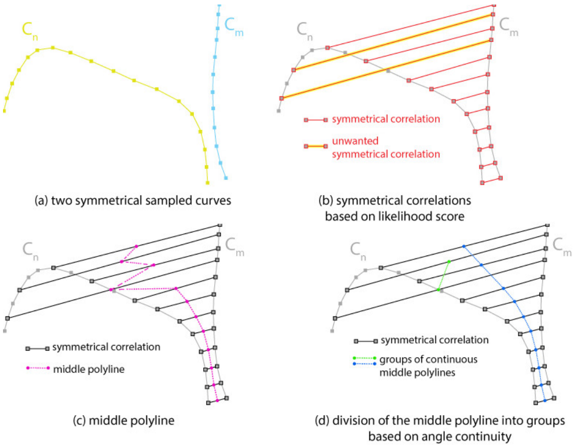

Figure 2: Finding consistent sets of symmetric points

polyline. We define the middle polyline of a subset of $S'_{m,n}$ as the line linking the middle points of the symmetrical correlations in the subset (cf Figure 2(c)). A middle polyline is considered continuous if the angle between each two successive segments of the polyline is sufficiently small. We finally chose $S_{m,n}$ as the subset of maximal cardinal among all the partitions of $S'_{m,n}$.

3.3. 3D curves reconstruction

We describe in the following the system of constraints we consider to compute the 3D coordinates of the input 2D curves. We note that our input data may contain several approximation. First, manual drawing on top of a picture may obviously contain positional approximations. Secondly, we assume the shape to be viewed from orthographic projection which is not true for real photography. Applying a direct reconstruction using only 2D positional and symmetry information as hard constraints as described in [CSMS13] could then lead to large discrepancies in 3D coordinates. Instead, we formulate our reconstruction algorithm as a set of smooth constraints, possibly non linear, which will be solved using a global optimization to ensure a more robust solution despite input approximations. More precisely, we consider the 3D coordinates of the control polygons of the curves as the unknown of the system, and express a set of constraints formulated as an energy functional to be minimized. For each 3D reconstructed point Q of the system, we will denote as $\bar{Q}$ the corresponding projection in the sketch.

3.3.1. Geometrical constraints

The system is constrained by 4 energy functional to be minimized. The first three constraints were presented in [XCS*14]. The last one, representing the feature of mirror-symmetry of the object, represents our main contribution.

Projection accuracy penalizes points P for which the projection onto the image plane $P|_{z=0}$ is far from the existing projection in the sketch $\bar{P}$: $E_{\text{proj}} = ||P|_{z=0} - \bar{P}||^2$

Minimal variation encodes the expectation that the coefficients relating points coordinate in 2D should not vary much in 3D. Take $\{Q_i\}_{i\in\{1..3\}}$ a control polygon, or a polygon made of the tangents 2 successive Bezier in a curve. Then, if $\{\bar{Q}_0, \bar{Q}_1, \bar{Q}_2\}$ are not collinear, we define:

$$E_{\text{minvar}} = ||q_0 Q_0 + q_1 Q_1 + q_2 Q_2 - Q_3||^2 ,$$

where $\bar{Q}_3 = q_0\bar{Q}_0 + q_1\bar{Q}_1 + q_2\bar{Q}_2$ and $q_0 + q_1 + q_2 = 1$. For each 3 control points $\{T_0, T_1, T_2\}$ collinear in the sketch, we define

$E_{\text{col}} = ||(1-t)T_0 - T_1 + tT_2||^2$, where $\bar{T}_1 = \bar{T}_0 + t\overrightarrow{\bar{T}_0\bar{T}_2}$.

Orthogonality enforces orthogonality of tangents $\overrightarrow{IQ_0}, \overrightarrow{IQ_1}$ for the intersections that were annotated by the user :

$$E_{\text{ortho}} = \left(\overrightarrow{IQ_0}.\overrightarrow{IQ_1}\right)^2$$

3.3.2. The Symmetry constraint

We present an energy functional enforcing the feature of mirror-symmetry in the sketch. We assume that the user has annotated which curves of the sketch are symmetric with each other. Self-symmetric curves are treated as the case of a pair of symmetric curves containing twice the same curve. To formulate this constraint, we use the result of the algorithm presented in Sec 3.2. For each pair of cubic Bezier (B^i, B^j), we get a set containing pairs of symmetrical sampled points $\{(V_k^i, V_l^j)\}_{k,l}$.

The first step of our approach is to compute the normal vector of the symmetry plane $n = (n_x, n_y, n_z)$. We denote as u, v the 3D orthogonal vectors in the symmetry plane whose projections are the annotated lines (green and cyan lines in Figure 1). Given the fact that $n|_{z=0} = (n_x, n_y)$ corresponds to the 2D line of symmetry, we solve : $\{u.v = 0,\ u.n = 0,\ v.n = 0\}$. We then follow the formulation of Cordier et al. [CSMS13] to define the symmetry constraints for all pairs (V_k^i, V_l^j). Specifically, we compute the 3D vector relating the two symmetrical points :

$$\overrightarrow{T}_{i,j,k,l} = \left[(v_k^i.x - v_l^j.x),\ (v_k^i.y - v_l^j.y),\ \frac{-n_z}{n_y}(v_k^i.y - v_l^j.y) \right]^{\mathrm{T}} \tag{3}$$

and their 3D middle point

$$\begin{aligned} M_{i,j,k,l} = \; & \tfrac{1}{2}\Big[(v_k^i.x + v_l^j.x),\ (v_k^i.y + v_l^j.y), \\ & -\left(\tfrac{n_x}{n_z}(v_k^i.x + v_l^j.x) + \tfrac{n_y}{n_z}(v_k^i.y + v_l^j.y) \right) \Big]^{\mathrm{T}} \end{aligned} \tag{4}$$

where $\bar{V}_k^i = [v_k^i.x,\ v_k^i.y]^{\mathrm{T}}$ and $\bar{V}_l^j = [v_l^j.x,\ v_l^j.y]^{\mathrm{T}}$.

We now need to express these constraints with respect to the 3D coordinates of the Bezier curves, which are the unknowns of our optimization. In 3D, each of the sampled points V_k^i can be expressed with a Bezier coordinate t_k^i such that

$$V_k^i = B^i(t_k^i),$$

where $B^i(t) = t^3 B_0^i + 3t(1-t)^2 B_1^i + 3t^2(1-t)B_2^i + t^3 B_3^i$.

With these notations, each symmetry correspondence (V_k^i, V_l^j) results in an energy on the Bezier control points

$$\begin{aligned} E_{\text{sym}} = \; & \left|\left| \left(B^i(t_k^i) - B^j(t_l^j) \right) - \overrightarrow{T}_{i,j,k,l} \right|\right|^2 \\ & + \left|\left| \tfrac{1}{2}\left(B^i(t_k^i) + B^j(t_l^j) \right) - M_{i,j,k,l} \right|\right|^2 . \end{aligned} \tag{5}$$

3.3.3. Energy functional

Finally, the different energy terms are assembled together in a global non linear energy function E

$$E = \omega_1 E_{\text{proj}} + \omega_2 E_{\text{minvar}} + \omega_3 E_{\text{col}} + \omega_4 E_{\text{ortho}} + \omega_5 E_{\text{sym}} .$$

In practice, we fix all weights to 1 except for $\omega_2 = 0.01$. The optimization is computed using a Newton's iterative method,

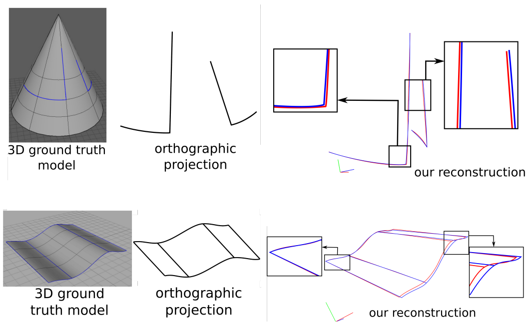

Figure 3: Validation : 3D curves are projected onto the image plane (left), and the third dimension is recovered with our algorithm (right). The results show small differences between the ground truth (in red) and our reconstruction (in blue) : 2% of error in average, with a variance smaller than 10^{-3}.

and initialized with the solution of a least square system over the symmetry constraint presented in Sec. 3.3.2.

3.4. Generation of developable surfaces

Once we recover a network of 3D curves representing the contours of the object, we decompose the network of 3D curves into closed cycles using the approach of Zhuang *et al.* [ZZCJ13]. Each cycle is then considered as the boundary of a developable surface, synthesized in the following way. First, a mesh interpolating the boundary and minimizing the dihedral angle between triangle is computed following the approach of Zhou *et al.* [ZJC13]. Second, the developability of each mesh is optimized by minimizing the Gaussian curvature at each vertex,using the local optimization algorithm presented by Wang *et al.* [WT04]. Finally, pattern pieces are computed using the flattening function of Blender.

4. Validation and results

Validation. We validated our method for 3D curve reconstruction on the test cases shown in Figure 3. These examples correspond to 3D Bezier surfaces created using Maya software and viewed from orthographic projection. Our reconstruction algorithm is then applied on the selected curves shown in blue on the figure, and compared with the true 3D coordinates of the curves. To perform a quantitative comparison, we introduce the *relative distance error* of a reconstructed point Q_i compared to its ground truth 3D coordinates $\tilde{Q}_i$ as $d_i = 1/D\,\|Q_i - \tilde{Q}_i\|$, where D is the length of the diagonal of the 3D model's bounding box.

Other results. We performed the reconstruction on different models. Once the curves are reconstructed, we apply the method presented in Sec. 3.4 to generate a piecewise developable surface. Results (see Figure 1, 4) show that plausible shapes can be reconstructed from the single photo, with around 1000 iterations of the optimization system. The maximal *angular defection*, measuring the lack of developability in a mesh and defined in [WT04], is, after optimization, at worst 1.2° for the cap model, and 2.1° for the bag model.

Limitations. The method raises several limitations. First, the symmetrical parts of the object may not remain symmetrically while reconstructed in the 3D space. Secondly, only the visible regions can be reconstructed. We aim at extending our approach by further using the symmetry information to fill-in the non visible parts of the object and improving

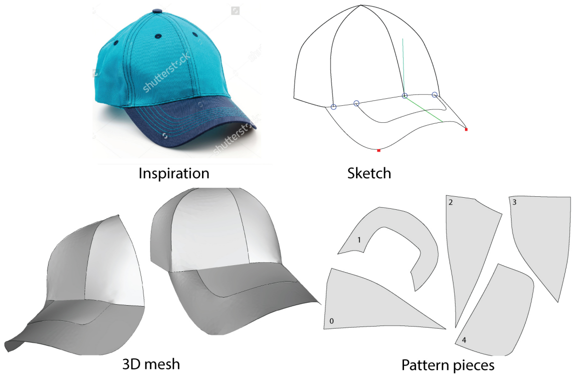

Figure 4: Another example of reconstruction : cap model.

the symmetry of the resulting surface. We also plan to extend our algorithm to more general shapes, including non fully symmetrical ones as well as developable surfaces that contain folds, as is common with soft fabric.

5. Conclusion

We described a method to model sewed objects from a single annotated photo. This method first recovers a network of 3D curves using non linear optimization based on geometrical assumptions: orthogonalities, symmetry and regularity of the curves. In a second step, developable surfaces are generated for each cycle in the curve network, and enable us to compute the corresponding 2D pattern pieces. We obtain good results for sketches that are described with sufficiently regular curves and represent the object from a general viewpoint.

References

[CSMS13] Cordier F., Seo H., Melkemi M., Sapidis N. S.: Inferring mirror symmetric 3D shapes from sketches. Computer-Aided Design. Vol. 45, Num. 2 (2013), 301–311.

[LFG08] Lee S., Feng D., Gooch B.: Automatic construction of 3D models from architectural line drawings. In Symp. on Interactive 3D Graphics and Games (2008), pp. 123–130.

[LS07] Lipson H., Shpitalni M.: Optimization-based reconstruction of a 3D object from a single freehand line drawing. In ACM SIGGRAPH Courses Notes (2007).

[OSSJ09] Olsen L., Samavati F. F., Sousa M. C., Jorge J. A.: Sketch-based modeling: A survey. CAG. Vol. 33 (2009).

[ÖUP*11] Öztireli A. C., Uyumaz U., Popa T., Sheffer A., Gross M.: 3d modeling with a symmetric sketch. In Sketch-Based Interfaces and Modeling (2011), pp. 23–30.

[SBSS12] Shao C., Bousseau A., Sheffer A., Singh K.: CrossShade: Shading Concept Sketches Using Cross-Section Curves. ACM ToG (Proc. SIGGRAPH). Vol. 31, Num. 4 (2012).

[WT04] Wang C. C., Tang K.: Achieving developability of a polygonal surface by minimum deformation: a study of global and local optimization approaches. The Visual Computer. Vol. 20, Num. 8-9 (2004), 521–539.

[XCS*14] Xu B., Chang W., Sheffer A., Bousseau A., McCrae J., Singh K.: True2Form: 3D curve networks from 2D sketches via selective regularization. ACM Transactions on Graphics (Proc. SIGGRAPH). Vol. 33, Num. 4 (2014).

[ZJC13] Zou M., Ju T., Carr N.: An algorithm for triangulating multiple 3D polygons. Symp. Geometry Processing (2013).

[ZZCJ13] Zhuang Y., Zou M., Carr N., Ju T.: A general and efficient method for finding cycles in 3D curve networks. ACM ToG (Proc. SIGGRAPH Asia). Vol. 32, Num. 6 (2013), 180.